Financial Modeling and Testing - Final Assignment

Building Portfolio Using News Articles

Harsh Mehta, Karthick Ramasubramanian, Rohitashwa Chakraborty

Executive Summary

We predicted the change in the direction of Net Income for next quarter using past 4 quarters of data. We took a long position in the stocks predicted to have a high probability of positive change in direction of net income and shorted the ones where there is low probability of net income going up, in other words high probability of net income going reverse . We also constructed a long only portfolio with stocks predicted to have a positive change in net income. Our best long-short portfolio had a Sharpe ratio of 2.4 and our best long only portfolio had a Sharpe ratio of 2.5.

Introduction

Our proposition for this project is simple. If the net income of a firm increases quarter over quarter, its share price will go up and vice versa. We have taken two steps back from percentage change in EPS prediction. First, we have taken a step back to pose this as a classification problem instead of a regression problem. We believe that while the % change in EPS might help us to decide the weights of stocks in the portfolio - higher % EPS change leads to higher weight, predicting the change in direction of EPS is enough to tell us which stocks to keep in the portfolio. Using probability cut-offs we can choose our own thresholds to pick our stocks for our portfolio. The reason for taking this step back is the low R-squared and high RMSE we observed in the models we built for predicting the % change in EPS. That is, the predictive power of our models was very low, hence subsequent decisions based on the output of our models had very low confidence. Consequently the portfolios we built using our predictions performed very poorly. The next step back we took was predicting the change in direction of Net Income instead of Earnings per Share. To get from NI to EPS we divide NI by the common shares outstanding. This step back involves making some assumptions that we believe are more than fair. We assume that from quarter to quarter, the common shares outstanding for a firm does not change significantly. While this assumption is violated in case of a merger or acquisition and stock split scenario, this information will be captured in our analysis of the key event and hence controlled for in our model. Moreover, this step back was necessary because of the low quality of data in the common shares outstanding column from WRDS. Hence taking this step back allows us to keep our target variable ‘pure’ without harming the relationship between it and the stock returns. We are doing this analysis for all the listed companies.

Data Sources

Edgar - Key Events

This database contains the news articles of all listed companies. The news articles are summarized, categorized and labeled. Categories include Earnings call, buybacks, M&As, Executive changes etc… We are aggregating this news to a quarter and therefore, not every company's news is present in every quarter. We are obtaining data from 2001 to 2020.

CRSP - Stock Prices

We are obtaining the quarterly stock price data for all listed companies from this database.

Compustat

All listed companies' fundamentals data from the Income and Balance Statement is obtained from this database. Variables include revenue, asset turnover, accruals, inventory, debt, operating income before depreciation and amortization etc…

Fama French Factors

We are getting the 3 Factor Fama French values and the risk free rate for every quarter from 2000 to 2020.

Everything aggregated we have data for more than 27000 companies over the period of 20 years. The model is tested on data from 2011 to 2020 and portfolios including the long only and long short are constructed on the same data.

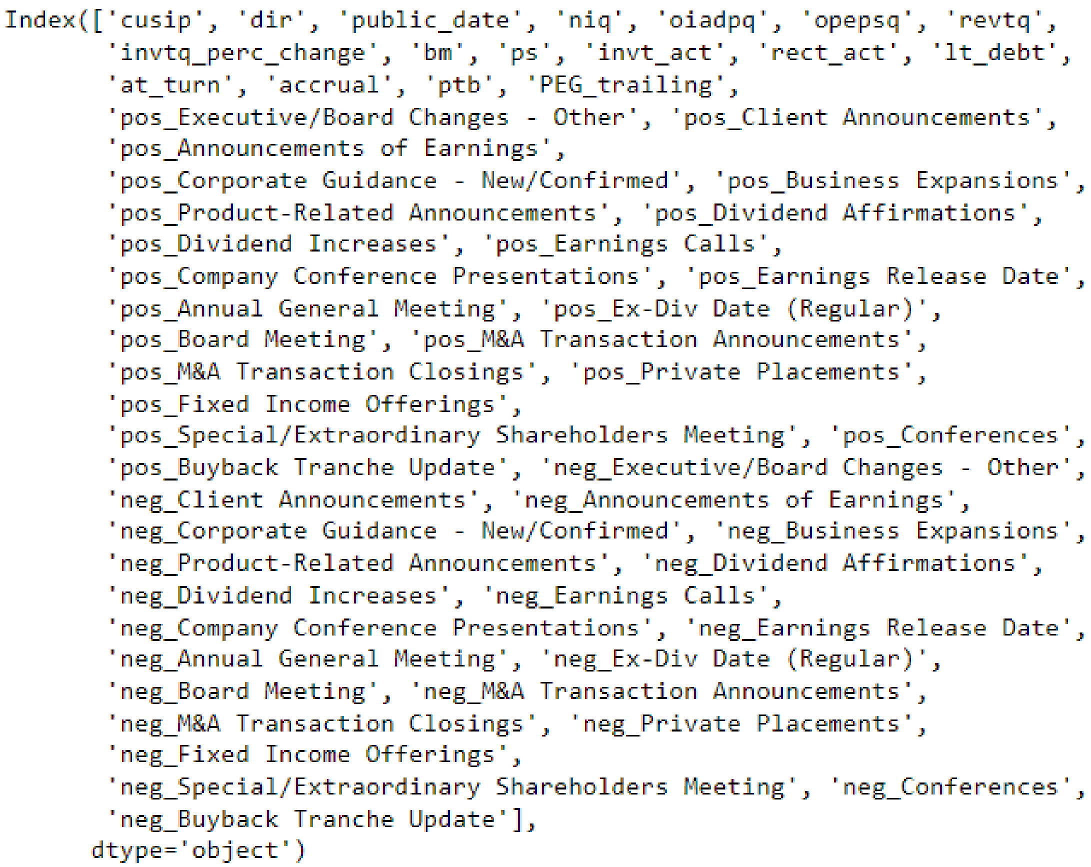
Analysis

Data Preparation

After pulling the firm fundamentals from the Compustat, we prepared our target variable for prediction. The dataset was grouped by GVKEY and the Net Income column was shifted by 1 row in each group. The shifted NI column was merged with the original dataset to calculate the direction of change in NI from one year to another for the particular company. Since our goal is to do supervised learning, we dropped all missing values in our target variable column.

The next step in the analysis was merging the firm financial ratios data with the fundamentals data. The date of availability of the financial ratio data was recorded in the public_date field of the dataset. We merged the two datasets on GVKEY and the appropriate date columns resulting in the data frame with all our structured data.

The unstructured data used in our models was pulled from the Key Events section of Edgar filings. The essential columns in the file included the data the event occurred, the event ID and a description of the event. Our objective was to analyze the description for sentiment and use that as a predictor in our model. The sentiment analysis is done using the VADER package in python. This algorithm gives a positive and negative for each paragraph/sentence. Also using the category of the news article, for every company for every quarter and every category we have 2 sentiment scores. By doing this, we exponentially increase the dimensions of our feature space but preserve all the data.



```
Index(['cusip', 'dir', 'public_date', 'niq', 'oiadpq', 'opepsq', 'revtq',
       'invtq_perc_change', 'bm', 'ps', 'invt_act', 'rect_act', 'lt_debt',
       'at_turn', 'accrual', 'ptb', 'PEG_trailing',
       'pos_Executive/Board Changes - Other', 'pos_Client Announcements',
       'pos_Announcements of Earnings',
       'pos_Corporate Guidance - New/Confirmed', 'pos_Business Expansions',
       'pos_Product-Related Announcements', 'pos_Dividend Affirmations',
       'pos_Dividend Increases', 'pos_Earnings Calls',
       'pos_Company Conference Presentations', 'pos_Earnings Release Date',
       'pos_Annual General Meeting', 'pos_Ex-Div Date (Regular)',
       'pos_Board Meeting', 'pos_M&A Transaction Announcements',
       'pos_M&A Transaction Closings', 'pos_Private Placements',
       'pos_Fixed Income Offerings',
       'pos_Special/Extraordinary Shareholders Meeting', 'pos_Conferences',
       'pos_Buyback Tranche Update', 'neg_Executive/Board Changes - Other',
       'neg_Client Announcements', 'neg_Announcements of Earnings',
       'neg_Corporate Guidance - New/Confirmed', 'neg_Business Expansions',
       'neg_Product-Related Announcements', 'neg_Dividend Affirmations',
       'neg_Dividend Increases', 'neg_Earnings Calls',
       'neg_Company Conference Presentations', 'neg_Earnings Release Date',
       'neg_Annual General Meeting', 'neg_Ex-Div Date (Regular)',
       'neg_Board Meeting', 'neg_M&A Transaction Announcements',
       'neg_M&A Transaction Closings', 'neg_Private Placements',
       'neg_Fixed Income Offerings',
       'neg_Special/Extraordinary Shareholders Meeting', 'neg_Conferences',
       'neg_Buyback Tranche Update'],
      dtype='object')
```

Feature Engineering

Some of the numeric columns in our dataset were read as String types because of the presence of non-numeric symbols like ‘%’. We used regex and lambda functions to clean these columns and convert them into the correct format. Then, for each of our features, we added a column to indicate the % change in the value of the feature from the last quarter. Our hypothesis is that both the level of the predictor and the change in the value of the predictor will be deemed important in our model.

We acknowledged the strong likelihood of multicollinearity in our dataset and to remove it as well as to select predictors for our model, we decided to do a regular stepwise regression. After giving thought to implementing Principal Component Analysis (PCA) we decided to not do it because we did not want to lose out on the interpretation of our predictors. We also wanted to avoid the risk of the components having low correlation with our target variable.

To find variables important to our analysis, we first ran univariate logistic regression with our target variable and each of our predictors. The predictors included the firm fundamentals, financial ratios along with the engineered % change columns for each of the predictors. The variables that were statistically significant in this simple regression model were passed on to the stepwise logistic regression model to evaluate their importance in the presence of the other features. Running a multivariate regression directly on all features would not give good results because of the large amount of missing values in our dataset. Running univariate regression first and then stepwise regression allows us to preserve as much data as possible while achieving our objective of selecting features important to our analysis.

We passed over a 100 features to the model and were left with 17 features. We analyzed the fields returned by the regression to ensure there is a plausible story behind them. For example, % change in inventory turnover was one of the statistically significant features in the regression. The relation between inventory turnover and Net Income is intuitive. Higher the turnover, higher the revenue and higher revenue signals higher net income. If a company is able to sell their inventory at a rate faster than their previous quarter's then it is likely that the revenue next quarter is higher than the revenue previous quarter along with the net income.

The features selected in the stepwise regression were used in our prediction using machine learning models.

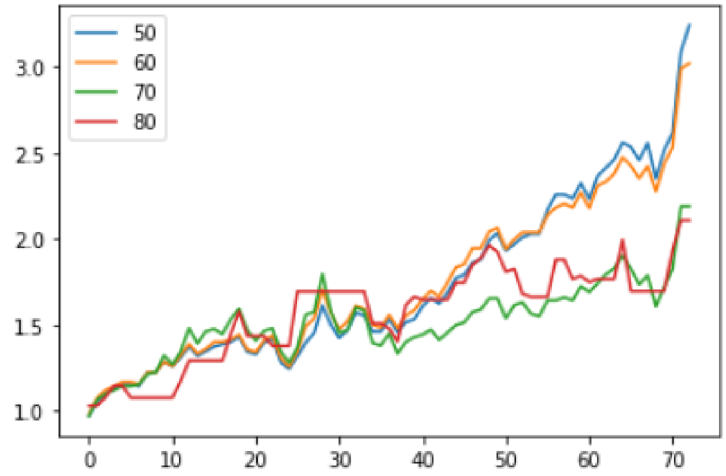
Modeling

To make our prediction, we trained our data using a Random Forest Classifier. The rationale for using this model includes past experience with earning prediction. Random Forest Classifier had the best area under the curve compared to both logistic regression and Xgboost.

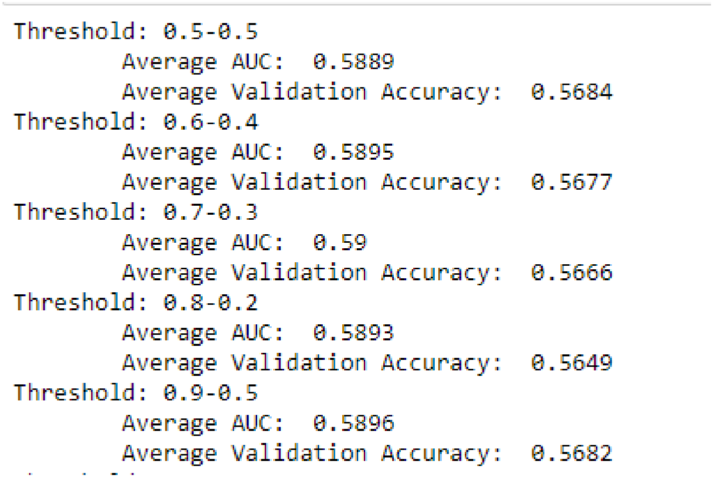
In this case, again the Random Forest Classifier performed the best. All models were tested out of sample with R2 values and AUC values. For the purpose of portfolio building, we used a rolling window for predictions. Meaning, for 2011, data from 2001 to 2010 was used, for 2012 data from 2001 to 2011 was used and so on.

We also used different probability thresholds to create our portfolios. The thresholds used were 50, 60, 70, 80 and 90. Meaning, only invest if the probability of %Change in Net Income going up is above the threshold.

Results



The above graph shows the portfolio returns based on creating a long short portfolio on different probability thresholds.



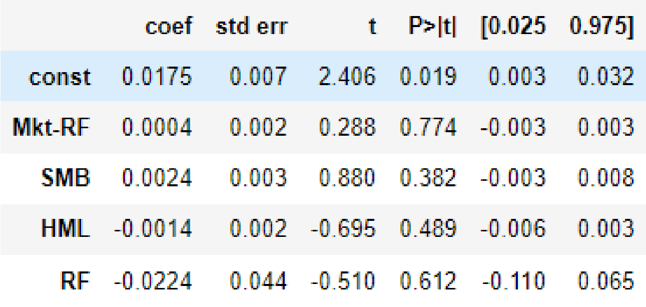
```
Threshold: 0.5-0.5
        Average AUC:  0.5889
        Average Validation Accuracy:  0.5684
Threshold: 0.6-0.4
        Average AUC:  0.5895
        Average Validation Accuracy:  0.5677
Threshold: 0.7-0.3
        Average AUC:  0.59
        Average Validation Accuracy:  0.5666
Threshold: 0.8-0.2
        Average AUC:  0.5893
        Average Validation Accuracy:  0.5649
Threshold: 0.9-0.5
        Average AUC:  0.5896
        Average Validation Accuracy:  0.5682
```

Based on the AUC values, portfolio with the 70% cutoff had the highest AUC, although other values were also pretty close.

For our final portfolio construction we went ahead with the 70% probability threshold cutoffs.

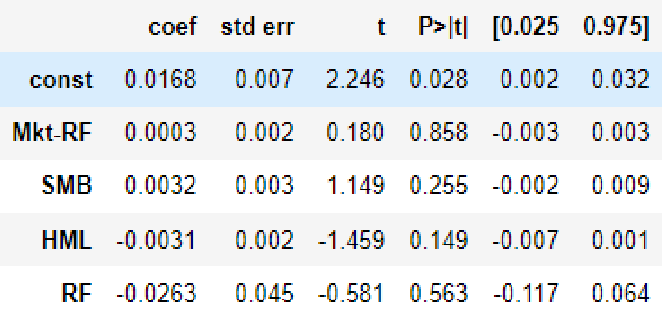
For the Long only portfolio, companies with over 70% chance of % Change in Net Income increasing were invested in. And for the Long-Short portfolio, the other companies were shorted to create a zero cost portfolio. We create an equal weighted portfolio which is balanced every quarter.

Long only Portfolio



	coef	std err	t	P>|t|	[0.025	0.975]
const	0.0175	0.007	2.406	0.019	0.003	0.032
Mkt-RF	0.0004	0.002	0.288	0.774	-0.003	0.003
SMB	0.0024	0.003	0.880	0.382	-0.003	0.008
HML	-0.0014	0.002	-0.695	0.489	-0.006	0.003
RF	-0.0224	0.044	-0.510	0.612	-0.110	0.065

Long Short Portfolio



	coef	std err	t	P>|t|	[0.025	0.975]
const	0.0168	0.007	2.246	0.028	0.002	0.032
Mkt-RF	0.0003	0.002	0.180	0.858	-0.003	0.003
SMB	0.0032	0.003	1.149	0.255	-0.002	0.009
HML	-0.0031	0.002	-1.459	0.149	-0.007	0.001
RF	-0.0263	0.045	-0.581	0.563	-0.117	0.064

For both portfolios, we observe a positive and statistically significant Alpha. Also, Betas (Market, value and size) are statistically essentially zeros.

With the above values, we obtain a sharpe ratio of 2.5 and 2.4 for Long Only and Long Short Portfolio respectively.

Conclusion

Our decision to look at only % Change in Net Income to eliminate granularity in our data has yielded very positive results. It has largely improved the performance of our machine learning models.

Getting a sharpe ratio that is much higher than 1 and building a portfolio with properly maintained data sources, we can say with high confidence that this is a good method to generate systematic low risk returns.